Amendments to the Title and Specification

Please replace the current title with the following amended title:

DATA PROCESSING DEVICE WITH A ~~SPARE FIELD IN THE INSTRUCTION~~PREDECODER OF THE INSTRUCTIONS

Please replace the paragraph at page 2, lines 1-12, with the following amended paragraph:

In recent years, to make processors operate at higher speeds, the pipeline stage is usually finely divided into individual stages and the logic steps in each of these stages are reduced to improve (raise) the operating frequency. To achieve ~~microprocessor~~ microprocessors for personal computers with a frequency higher than 1 gigahertz (GHz), a microarchitecture (super pipeline method) may for example be defined having a pipeline stage made up of dozens of stages. However, when the number of pipeline stages are increased and a branching prediction error occurs during branching, an extremely large penalty accompanies this error.

Please replace the paragraph at page 2, lines 13-21, with the following amended paragraph:

The inventors studied how to reduce this kind of penalty. Speeding up the decoding and execution of instructions for example for branch instructions is effective in reducing these kinds of penalties. This speedup can also be achieved by adding new instructions, and by renewing the instruction set however problems occur. It is ~~because,~~ because there is a strong need to keep using the existing

software even if the computer hardware has become more advanced and so upward compatibility is required.

Please replace the paragraph at page 5, lines 6-25, with the following amended paragraph:

The control means at this time is for the split-branch method. More specifically, a control means for the split-branch method possesses a queuing buffer to temporarily hold the instruction loaded from the instruction cache memory, possesses an instruction set holding a pre-branch processing instruction and branch processing instruction for dividing up one branching operation, (this pre-branch processing instruction is for commanding calculation of the branch destination address and fetching of the branch destination instruction) and possesses a target buffer for temporarily holding a branch destination instruction and branch destination address acquired from executing the branch destination processing instruction. When the control means ~~had~~ has determined that the instruction is a branch processing instruction by means of information for the area corresponding to the specified field of the instruction held in the queuing ~~buffer;~~ buffer, then the control means issues instructions to load the branch instruction and the subsequent branch destination address from the target buffer.

Please replace the paragraph at page 13, line 13 – page 14, line 4, with the following amended paragraph:

The operation of the predecode-processor (PD) 100 according to the second aspect of the function expansion is explained next. The branch instruction loaded from the external memory 106 is supplied to the predecode-processor (PD) 100 from the ~~BIU102~~ BIU 102. The operation code ~~op121~~ (op) 121 is decoded by the predecoder 130 inside the predecode-processor (PD) 100 and a decision is made only whether this instruction is a branch instruction or not. If it decides that the instruction is a branch instruction than a "1" is set in the output rsv[1] of predecoder 130 so this instruction can be identified as a branch instruction. The output rsv[1] = "1" is then stored in a field in the instruction cache memory 101 corresponding to the spare field 127 of the applicable instruction. Only a branch instruction was selected in the example given here, however, the selection is not limited to branch instructions and the designer can select an instruction as needed.